**Tourism Management System**



Session: 2022 – 2025

**Submitted by:**

samiullah 2022-cs-82

**Supervised by:**

muhammad irzam

Department of Computer Science

**University of Engineering and Technology**

**Lahore Pakistan**

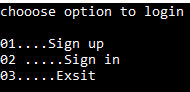
**Here you can find the major parts of your Business Application documentation**

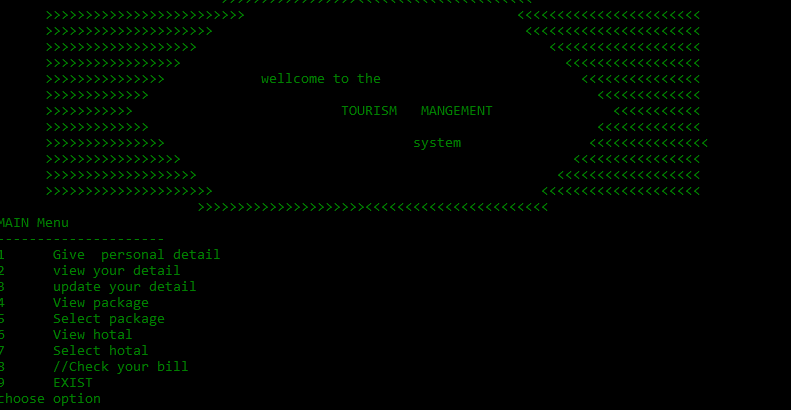
* **Short Description of your project**
  + in this system we see package for tourism and select package .Same we do this for hotel . And at the end system generate bill of all detail of user and package and detail
* **Users of Application (minimum 2 users for your project)**
  + in this system there are two user
  + first is admin and other is customer
* **Functional Requirements**
  + Functional requirements are a detailed description of what a system, product, or service must do to meet the needs and expectations
  + Use the following format to write the Functional Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| User Story ID | *As a* | *I want to perform* | *So that I can* |
| **1** | Admin |  | Calculate the Price of product w.r.t. it's quantity. |

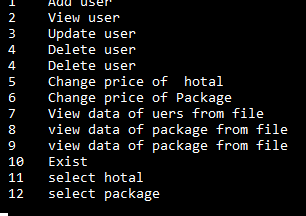
* **Wireframes**
  + Actual Wireframes of your business Application
  + For Example

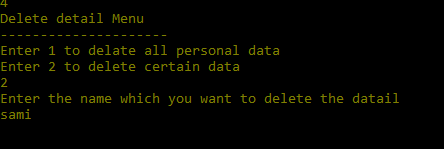
**Figure 1: LogIn Screen**

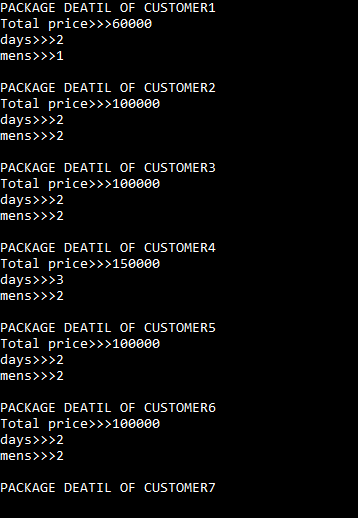
****

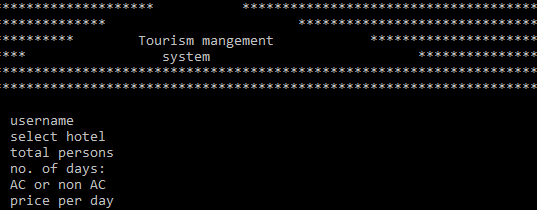


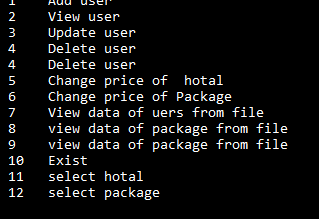


****

****

****

****

****

**Important Instructions**

**Figure 2: Admin Main Menu Screen**

* **Data Structures (Parallel Arrays)**
* **string name, gender, phone, anotherphone, nationality, email;**
* **string nameA[100], genderA[100], phoneA[100], anotherphoneA[100], nationalityA[100], emailA[100];**
* **int b, option\_admin, count, s = 0, y = 0, x = 0, optionpackage, days, mens, daysA[100], mensA[100];**
* **string username, username1, role1, role2;**
* **string password, password1;**
* **int package\_1\_price = 30000, package\_2\_price = 25000, package\_3\_price = 20000, count\_change\_price\_package;**
* **int total\_price\_of\_packages[100];**
* **Function Prototypes**

int usermanu();

void adduer\_into\_Array(string name, string gender, string phone, string anotherphone, string nationality, string email);

void viewuser\_deatil();

void adduser();

void package();

void selectpakage();

void calculation\_of\_packages(int days, int mens);

void view\_all\_data();

void about();

void sign\_in\_sign\_up();

void update\_selected\_package();

void calculation\_of\_packages(int days, int mens);

int admin\_menu();

void change\_the\_price\_of\_package();

void userheader();

void add\_user\_haeder();

void view\_user\_header();

void updated\_user\_detail();

void delete\_user\_detail();

void add\_user\_into\_file();

void packges\_deatil\_into\_file();

void update\_adduser\_detail\_into\_file();

void delete\_user\_detail\_into\_file();

void subMenuBeforeMainMenu(string submenu);

void READ\_DATA\_from\_file\_of\_add\_user();

string getfield(string line, int field);

string getfield1(string line1, int field1);

string getfield2(string line2, int field2);

string usernames[100];

string passwords[100];

int usercount = 0;

bool signIn(string username, string password);

void addUsers(string username, string password);

void viewUsers();

void writeFile(string username, string password);

bool isValidUsername(string username);

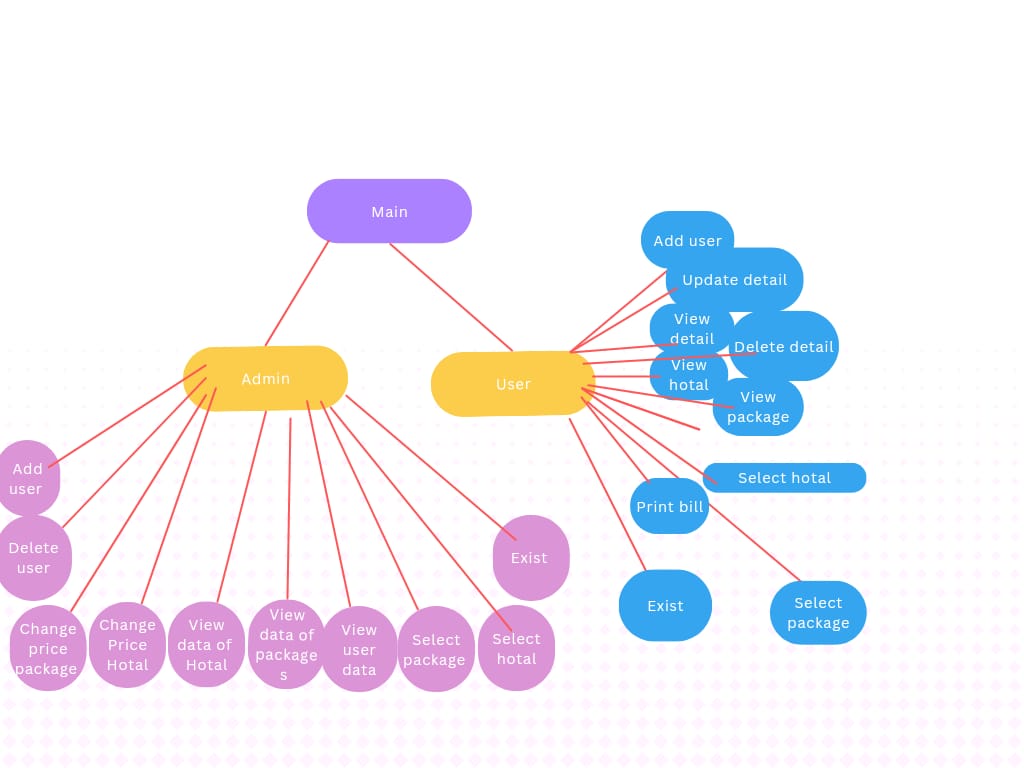
int takeChoice();

void loadDataFromFile();

//bill

void bill();

* **Functions Working Flow**

****

* **Complete Code of the Business Application**
* #include <iostream>
* #include <conio.h>
* #include <windows.h>
* #include <fstream>
* #include <iomanip>
* #include <conio.h>
* using namespace std;
* string admin\_user();
* int usermanu();
* void adduer\_into\_Array(string name, string gender, string phone, string anotherphone, string nationality, string email);
* void viewuser\_deatil();
* void adduser();
* void package();
* void selectpakage();
* void calculation\_of\_packages(int days, int mens);
* void view\_all\_data();
* void about();
* void sign\_in\_sign\_up();
* void update\_selected\_package();
* void calculation\_of\_packages(int days, int mens);
* int admin\_menu();
* void change\_the\_price\_of\_package();
* void userheader();
* void add\_user\_haeder();
* void view\_user\_header();
* void updated\_user\_detail();
* void delete\_user\_detail();
* void add\_user\_into\_file();
* void packges\_deatil\_into\_file();
* void update\_adduser\_detail\_into\_file();
* void delete\_user\_detail\_into\_file();
* void subMenuBeforeMainMenu(string submenu);
* void READ\_DATA\_from\_file\_of\_add\_user();
* string getfield(string line, int field);
* string getfield1(string line1, int field1);
* string getfield2(string line2, int field2);
* // hotal
* void hotel();
* void select\_hoatl();
* void calulation\_of\_hotal\_deatil(int days, int mens, int number\_for\_AC\_or\_non\_AC, int choice);
* void store\_data\_of\_hotal\_deatil\_into\_file();
* int option\_hotals, count\_for\_AC\_or\_non\_AC;
* int hotal\_mens;
* int hotal\_days;
* int hotal\_mensA[100];
* int hotal\_daysA[100];
* int total\_price\_of\_hotal[100];
* int price\_of\_luxury\_hotal = 20000, price\_of\_middle\_class\_hoatl = 15000, price\_of\_normal\_class\_hoatl = 10000;
* int hotal\_index = 0;
* void view\_hotal();
* int count\_for\_show\_hotal\_detail = 0;
* void show\_record\_hotals(string tth, string dsh, string msh);
* void READ\_DATA\_from\_file\_of\_hotals();
* // package
* string getfield1(string line1, int field1);
* void show\_record\_packages(string tt, string ds, string ms);
* void READ\_DATA\_from\_file\_of\_packages();
* int count\_for\_show\_package\_deatil\_from\_file = 0;
* // preference variable for hotal and packages
* int preference\_variable = 0;
* // change the price of hotal
* void changr\_the\_price\_of\_hotal();
* int m, n, o;
* // delete data of user
* string name\_checking;
* // change the price of hotal
* void shange\_the\_price\_of\_packages();
* int p, q, r;
* // admin user
* int count\_student = 0;
* //
* int a = 0;
* int idx = 0;
* //
* string name, gender, phone, anotherphone, nationality, email;
* string nameA[100], genderA[100], phoneA[100], anotherphoneA[100], nationalityA[100], emailA[100];
* int b, option\_admin, count, s = 0, y = 0, x = 0, optionpackage, days, mens, daysA[100], mensA[100];
* string username, username1, role1, role2;
* string password, password1;
* int package\_1\_price = 30000, package\_2\_price = 25000, package\_3\_price = 20000, count\_change\_price\_package;
* int total\_price\_of\_packages[100];
* string showbase\_file\_record(string name1, string gender1, string phone1, string an\_phone, string nationality1, string email1);
* int count\_for\_show\_user\_deatil\_from\_file = 0;
* /// signup
* string usernames[100];
* string passwords[100];
* int usercount = 0;
* bool signIn(string username, string password);
* void addUsers(string username, string password);
* void viewUsers();
* void writeFile(string username, string password);
* bool isValidUsername(string username);
* int takeChoice();
* void loadDataFromFile();
* //bill
* void bill();
* ///
* int main()
* {






* while (true)
* {
* system("cls");
* system("Color 0F");
* sign\_in\_sign\_up();
* if (count == 2)
* {
* system("cls");
* cout << "Enter the username" << endl;
* getline(cin.ignore(), username1);
* cout << "Enter the password" << endl;
* getline(cin.ignore(), password1);
* getch();
* }
* if (count == 3)
* {
* break;
* }
* string m = admin\_user();
* cout << m;
* if (m == "user")
* {
* system("Color 09");
* while (true)
* {
* a = usermanu();
* if (a == 1)
* {
* adduser();
* }
* else if (a == 2)
* {
* viewuser\_deatil();
* }
* else if (a == 3)
* {
* updated\_user\_detail();
* }
* else if (a == 4)
* {
* package();
* }
* else if (a == 5)
* {
* selectpakage();
* }
* else if (a == 6)
* {
* view\_hotal();
* }
* else if (a == 7)
* {
* select\_hoatl();
* }
* else if (a == 8)
* {
* bill();
* getch();
* }
* else if (a == 9)
* {
* break;
* }
* // else if (a == 10)
* // {
* //     READ\_DATA\_from\_file\_of\_add\_user();
* //     for (int x = 0; x < 1; x++)
* //     {
* //         cout << "Customer no. " << x + 1 << endl;
* //         cout << nameA[x] << endl;
* //         cout << genderA[x] << endl;
* //         cout << phoneA[x] << endl;
* //         cout << anotherphoneA[x] << endl;
* //         cout << nationalityA[x] << endl;
* //         cout << emailA[x] << endl;
* //     }
* //     getch();
* // }
* // else if (a == 11)
* // {
* //     select\_hoatl();
* // }
* // else if (a == 12)
* // {
* //     cout << total\_price\_of\_hotal[0] << endl;
* //     getch();
* // }
* // else if (a == 13)
* // {
* //     changr\_the\_price\_of\_hotal();
* //     getch();
* // }
* else
* {
* cout << "you press wrong" << endl;
* }
* }
* }
* else if (m == "admin")
* {
* while (true)
* {
* b = admin\_menu();
* if (b == 1)
* {
* adduser();
* }
* else if (b == 2)
* {
* viewuser\_deatil();
* }
* else if (b == 3)
* {
* updated\_user\_detail();
* }
* else if (b == 4)
* {
* delete\_user\_detail();
* }
* else if (b == 5)
* {
* changr\_the\_price\_of\_hotal();
* }
* else if (b == 6)
* {
* change\_the\_price\_of\_package();
* }
* else if (b == 7)
* {
* READ\_DATA\_from\_file\_of\_add\_user();

* getch();
* }
* else if (b == 8)
* {
* READ\_DATA\_from\_file\_of\_packages();
* getch();
* }
* else if (b == 9)
* {
* READ\_DATA\_from\_file\_of\_hotals();
* getch();
* }
* else if (b == 10)
* {
* break;
* }
* else if (b == 11)
* {
* select\_hoatl();
* }
* else if (b == 12)
* {
* selectpakage();
* }
* else
* {
* cout << "you press wrong " << endl;
* }
* }
* }
* }
* }
* string admin\_user()
* {
* system("cls");
* cout << "now you are in signup page" << endl;
* Sleep(500);
* cout << "Enter the username" << endl;
* cin >> username;
* cout << "Enter the password" << endl;
* cin >> password;
* getch();
* Sleep(400);
* fstream data;
* data.open("fileforsignip.txt",ios::app);
* data<<username;
* data<<",";
* data<<password;
* data.close();
* if (username == "sami" && password == "456")
* {
* return "user";
* }
* else if (username == "sami" && password == "123")
* {
* return "admin";
* }
* else
* {
* cout << "wrong information" << endl;
* }
* }
* int usermanu()
* {
* system("cls");
* userheader();
* subMenuBeforeMainMenu("MAIN");
* cout << "1      Give  personal detail" << endl;
* cout << "2      view your detail" << endl;
* cout << "3      update your detail   " << endl;
* cout << "4      View package      " << endl;
* cout << "5      Select package     " << endl;
* cout << "6      View hotal    " << endl;
* cout << "7      Select hotal " << endl;
* cout << "8      //Check your bill      " << endl;
* cout << "9      EXIST        " << endl;
* int option;
* cout << "choose option" << endl;
* cin >> option;
* return option;
* }
* void adduser()
* {
* system("cls");
* add\_user\_haeder();
* subMenuBeforeMainMenu("Add your detail");
* if (count\_student == hotal\_index)
* {
* cout << "  Enter your name        " << endl;
* cin.clear();
* cin.sync();
* getline(cin, name);
* cout << "  Enter your  nationality    " << endl;
* cin.clear();
* cin.sync();
* getline(cin, nationality);
* cout << " Enter the Gender " << endl;
* cin.clear();
* cin.sync();
* getline(cin, gender);
* cout << " Enter the e mainl " << endl;
* cin.clear();
* cin.sync();
* getline(cin, email);
* cout << " Enter the phone no. " << endl;
* cin.clear();
* cin.sync();
* getline(cin, phone);
* cout << " Enter the another phonenumber " << endl;
* cin.clear();
* cin.sync();
* getline(cin, anotherphone);
* adduer\_into\_Array(name, gender, phone, anotherphone, nationality, email);
* }
* else if (count\_student != hotal\_index)
* {
* cout << "Please first enter the hotal data and package " << endl;
* cout << " Then your are able to add new user" << endl;
* getch();
* }
* }
* void adduer\_into\_Array(string name, string gender, string phone, string anotherphone, string nationality, string email)
* {
* nameA[count\_student] = name;
* genderA[count\_student] = gender;
* phoneA[count\_student] = phone;
* anotherphoneA[count\_student] = anotherphone;
* nationalityA[count\_student] = nationality;
* emailA[count\_student] = email;
* count\_student++;
* add\_user\_into\_file();
* }
* void viewuser\_deatil()
* {
* system("cls");
* view\_user\_header();
* subMenuBeforeMainMenu("View your detail");
* //  READ\_DATA\_from\_file();
* if (count\_student == 0)
* {
* cout << "please enter first data" << endl;
* getch();
* }
* else
* {
* string view\_user\_deatil\_name;
* cout << "enter name which you want to see detail" << endl;
* cin >> view\_user\_deatil\_name;
* int number = 0;
* for (int x = 0; x < count\_student; x++)
* {
* if (view\_user\_deatil\_name == nameA[x])
* {
* number = 2;
* cout << "name is  >>>>>>>>>>>>>>>" << nameA[x] << endl;
* cout << "gender is >>>>>>>>>>>>>" << genderA[x] << endl;
* cout << "phone no. is >>>>>>>>>>>" << phoneA[x] << endl;
* cout << "second phone no. is >>>>" << anotherphoneA[x] << endl;
* cout << "nationality is >>>>>>>>>" << nationalityA[x] << endl;
* cout << "your email is >>>>>>>>>> " << emailA[x] << endl;
* cout << "      " << endl
* << endl
* << endl;
* cout << "Data successfully searched " << endl;
* getch();
* }
* }
* if (number != 2)
* {
* cout << "you press wrong " << endl;
* getch();
* }
* }
* }
* void package()
* {
* system("cls");
* subMenuBeforeMainMenu("View package");
* cout << "1.Golden package" << endl;
* cout << package\_1\_price << "Rs." << endl;
* cout << "2. Middle package" << endl;
* cout << package\_2\_price << "Rs." << endl;
* cout << "3. Normal package" << endl;
* cout << package\_3\_price << "Rs." << endl;
* getch();
* }
* void selectpakage()
* {
* subMenuBeforeMainMenu("Select package");
* if (count\_student == 0)
* {
* cout << "First enter the data" << endl;
* getch();
* }
* else
* {
* string name;
* cout << "Enter the name " << endl;
* cin >> name;
* int c = 0;
* for (int x = 0; x < count\_student; x++)
* {
* if (name == nameA[x])
* {
* c = 2;
* cout << "which is  your  favourite package press anyone from these(1,2,3)" << endl;
* cout << "1 Golden package" << endl;
* cout << "2 middle package" << endl;
* cout << "3 lower package" << endl;
* cin >> optionpackage;
* // if (optionpackage != 1 || optionpackage != 2 || optionpackage != 3)
* // {
* //     cout << "you press wrong " << endl;
* //     getch();
* //     break;
* // }
* cout << " Enter the days" << endl;
* cin >> days;
* cout << " Enter the members" << endl;
* cin >> mens;
* calculation\_of\_packages(days, mens);
* }
* }
* // if (c == 2)
* // {
* //     cout << "package is successfully selected" << endl;
* // }
* if (c == 0)
* {
* cout << "you press wrong" << endl;
* }
* getch();
* }
* }
* void calculation\_of\_packages(int days, int mens)
* {
* daysA[y] = days;
* mensA[y] = mens;
* if (optionpackage == 1)
* {
* total\_price\_of\_packages[y] = daysA[y] \* mensA[y] \* package\_1\_price;
* }
* else if (optionpackage == 2)
* {
* total\_price\_of\_packages[y] = daysA[y] \* mensA[y] \* package\_2\_price;
* }
* else if (optionpackage == 3)
* {
* total\_price\_of\_packages[y] = daysA[y] \* mensA[y] \* package\_3\_price;
* }
* else
* {
* cout << "you press wrong" << endl;
* }
* packges\_deatil\_into\_file();
* y++;
* }
* void view\_all\_data()
* {
* subMenuBeforeMainMenu("View your all data");
* if (y == 0)
* {
* cout << "First enter the data about package " << endl;
* getch();
* }
* else
* {
* string name;
* cout << "Enter the name for which you see the information about package" << endl;
* cin >> name;
* int z = 0;
* for (int x = 0; x < count\_student; x++)
* {
* if (name == nameA[x])
* {
* z = 2;
* cout << "The total amount of the package is>>>>>>>>>>>>  " << endl
* << total\_price\_of\_packages[x] << endl;
* cout << " for " << mensA[x] << "mens" << endl;
* cout << " for " << daysA[x] << "days" << endl;
* getch();
* }
* }
* if (z != 2)
* {
* cout << "you press wrong " << endl;
* }
* getch();
* }
* }
* void about()
* {
* cout << "this application is about tourrism.. in thsi application we select hotal and packages" << endl;
* getch;
* }
* void sign\_in\_sign\_up()
* {
* cout << "chooose option to login           " << endl;
* cout << "           " << endl;
* cout << "01....Sign up           " << endl;
* cout << "02 .....Sign in           " << endl;
* cout << "03.....Exsit           " << endl;
* cout << "   " << endl;
* cin >> count;
* }
* void update\_selected\_package()
* {
* days = 0;
* mens = 0;
* calculation\_of\_packages(days, mens);
* cout << " Enter the days" << endl;
* cin >> days;
* cout << " Enter the members" << endl;
* cin >> mens;
* calculation\_of\_packages(days, mens);
* }
* int admin\_menu()
* {
* system("cls");
* cout << " 1    Add user " << endl;
* cout << " 2    View user " << endl;
* cout << " 3    Update user " << endl;
* cout << " 4    Delete user " << endl;
* cout << " 4    Delete user " << endl;
* cout << " 5    Change price of  hotal" << endl;
* cout << " 6    Change price of Package" << endl;
* cout << " 7    View data of uers from file" << endl;
* cout << " 8    view data of package from file" << endl;
* cout << " 9    view data of package from file " << endl;
* cout << " 10   Exist  " << endl;
* cout << " 11   select hotal " << endl;
* cout << " 12   select package " << endl;
* cin >> option\_admin;
* return option\_admin;
* }
* void change\_the\_price\_of\_package()
* {
* cout << "These the prices of the packages" << endl;
* cout << "The price of package one is " << package\_1\_price << endl;
* cout << "The price of package two is " << package\_2\_price << endl;
* cout << "The price of package three is " << package\_3\_price << endl;
* cout << " " << endl
* << " " << endl;
* cout << "if you want to change the prices of packages  " << endl;
* cout << "please press 1" << endl;
* cin >> count\_change\_price\_package;
* if (count\_change\_price\_package == 1)
* {
* cout << "Enter the new price of package 1 " << endl;
* cin >> package\_1\_price;
* cout << "Enter the new price of package 2 " << endl;
* cin >> package\_2\_price;
* cout << "Enter the new price of package 3 " << endl;
* cin >> package\_3\_price;
* }
* else
* {
* cout << "You press wrong " << endl;
* }
* getch();
* }
* void userheader()
* {
* system("Color 02");
* cout << "                            >>>>>>>>>>>>>>>>><<<<<<<<<<<<<<<<<<<<<<              " << endl;
* cout << "      >>>>>>>>>>>>>>>>>>>>>>>>>                                  <<<<<<<<<<<<<<<<<<<<<<<           " << endl;
* cout << "      >>>>>>>>>>>>>>>>>>>>>                                       <<<<<<<<<<<<<<<<<<<<<<             " << endl;
* cout << "      >>>>>>>>>>>>>>>>>>>                                            <<<<<<<<<<<<<<<<<<<              " << endl;
* cout << "      >>>>>>>>>>>>>>>>>                                                <<<<<<<<<<<<<<<<<                 " << endl;
* cout << "      >>>>>>>>>>>>>>>            wellcome to the                         <<<<<<<<<<<<<<<               " << endl;
* cout << "      >>>>>>>>>>>>>                                                        <<<<<<<<<<<<<                   " << endl;
* cout << "      >>>>>>>>>>>                          TOURISM   MANGEMENT               <<<<<<<<<<<                " << endl;
* cout << "      >>>>>>>>>>>>>                                                        <<<<<<<<<<<<<          " << endl;
* cout << "      >>>>>>>>>>>>>>>                               system                <<<<<<<<<<<<<<<                              " << endl;
* cout << "      >>>>>>>>>>>>>>>>>                                                 <<<<<<<<<<<<<<<<" << endl;
* cout << "      >>>>>>>>>>>>>>>>>>>                                             <<<<<<<<<<<<<<<<<<" << endl;
* cout << "      >>>>>>>>>>>>>>>>>>>>>                                         <<<<<<<<<<<<<<<<<<<<" << endl;
* cout << "                         >>>>>>>>>>>>>>>>>>>>><<<<<<<<<<<<<<<<<<<<<<<                          " << endl;
* }
* void add\_user\_haeder()
* {
* system("Color 06");
* cout << "            \_      \_                                       \_          \_\_           " << endl;
* cout << "           | |    | |                                     (\_)        / \_|         " << endl;
* cout << "  \_\_ \_   \_\_| |  \_\_| |       \_   \_   \_\_\_   \_   \_  \_ \_\_      \_  \_ \_\_  | |\_  \_\_\_     " << endl;
* cout << " / \_` | / \_` | / \_` |      | | | | / \_  || | | || '\_\_|    | || '\_ . |  \_|/ \_  )  " << endl;
* cout << "| (\_| || (\_| || (\_| |      | |\_| || (\_) || |\_| || |       | || | | || | | (\_) |\_  " << endl;
* cout << " (\_\_,\_| (\_\_,\_| (\_\_,\_|       (\_\_, | ( \_\_\_/  (\_\_,\_||\_|       |\_||\_| |\_||\_| (\_\_\_/(\_) " << endl;
* cout << "                             \_\_/ |                                                " << endl;
* }
* void view\_user\_header()
* {
* cout << "      \_\_  \_\_    \_\_\_         " << endl;
* cout << " (|) |\_\_)(  )|  ||    |  |  " << endl;
* cout << "# ~ #|\_\_)(\_\_)(\_\_)|    (\_\_)  " << endl;
* cout << "                            " << endl;
* cout << "                            " << endl;
* cout << "                            " << endl;
* }
* void updated\_user\_detail()
* {
* subMenuBeforeMainMenu("Update detail");
* system("cls");
* string name1;
* cout << "Enter you name to update " << endl;
* cin >> name1;
* bool checking = false;
* for (int x = 0; x < count\_student; x++)
* {
* if (name1 == nameA[x])
* {
* checking = true;
* cout << "  Enter your name        " << endl;
* cin >> nameA[x];
* cout << "  Enter your  nationality    " << endl;
* cin >> nationalityA[x];
* cout << " Enter the Gender " << endl;
* cin >> genderA[x];
* cout << " Enter the e mainl " << endl;
* cin >> emailA[x];
* cout << " Enter the phone no. " << endl;
* cin >> phoneA[x];
* cout << " Enter the another phonenumber " << endl;
* cin >> anotherphoneA[x];
* cout << "you successfully updates you information" << endl
* << endl;
* update\_adduser\_detail\_into\_file();
* }
* }
* if (checking != true)
* {
* cout << "you enter wrong " << endl;
* }
* getch();
* }
* void delete\_user\_detail()
* {
* subMenuBeforeMainMenu("Delete detail");
* if (count\_student = 0)
* {
* cout << "please first data enter" << endl;
* }
* else
* {
* int b;
* cout << "Enter 1 to delate all personal data" << endl;
* cout << "Enter 2 to delete certain data" << endl;
* cin >> b;
* if (b == 1)
* {
* count\_student = 0;
* cout << "Data deleted completely successful" << endl;
* }
* else if (b = 2)
* {
* cout << "Enter the name which you want to delete the datail " << endl;
* cin >> name\_checking;
* int c = 0;
* for (int x = 0; x < count\_student; x++)
* {
* if (name\_checking == nameA[x])
* {
* for (int j = x; j < count\_student; j++)
* {
* c = 2;
* nameA[j] = nameA[j + 1];
* genderA[j] = genderA[j + 1];
* phoneA[j] = phoneA[j + 1];
* anotherphoneA[j] = anotherphoneA[j + 1];
* nationalityA[j] = nationalityA[j + 1];
* emailA[j] = emailA[j + 1];
* }
* count\_student--;
* cout << "Your data delete successfully" << endl;
* getch();
* }
* if (c != 2)
* {
* cout << "     you are data not deleted,press wrong" << endl;
* getch();
* }
* }
* delete\_user\_detail\_into\_file();
* }
* else
* {
* cout << "you press wrong " << endl;
* getch();
* }
* }
* }
* void add\_user\_into\_file()
* {
* fstream file;
* file.open("filenumber01.txt", ios::out);
* for (int x = 0; nameA[x] != ""; x++)
* {
* file << nameA[x];
* file << ",";
* file << genderA[x];
* file << ",";
* file << phoneA[x];
* file << ",";
* file << anotherphoneA[x];
* file << ",";
* file << nationalityA[x];
* file << ",";
* file << emailA[x];
* file << ",";
* file << endl;
* }
* file.close();
* }
* void packges\_deatil\_into\_file()
* {
* fstream file1;
* file1.open("filenumber02.txt", ios::app);
* file1 << "Total price>>>" << total\_price\_of\_packages[y];
* file1 << ",";
* file1 << "days>>>" << daysA[y];
* file1 << ",";
* file1 << "mens>>>" << mensA[y];
* file1 << ",";
* file1 << endl;
* file1.close();
* }
* void update\_adduser\_detail\_into\_file()
* {
* fstream file;
* file.open("filenumber01.txt", ios::out);
* for (int x = 0; x < count\_student; x++)
* {
* file << "name>>" << nameA[x];
* file << " ";
* file << "gender>" << genderA[x];
* file << " ";
* file << "phone>>" << phoneA[x];
* file << " ";
* file << "anotherphone>>" << anotherphoneA[x];
* file << " ";
* file << "nationality>>" << nationalityA[x];
* file << " ";
* file << "email>>" << emailA[x];
* file << " ";
* file << endl
* << endl;
* }
* file.close();
* }
* void delete\_user\_detail\_into\_file()
* {
* string line;
* fstream file;
* file.open("filenumber01.txt", ios::out);
* // for (int j = 0; j < 1; j++)
* int j = 0;
* while (!(file.eof()))
* {
* if (name\_checking == nameA[j])
* {
* cout << j << endl;
* } // getline(file,line);
* // if (name\_checking == nameA[j])
* // {
* //     nameA[j] = nameA[j + 1];
* //     genderA[j] = genderA[j + 1];
* //     phoneA[j] =   phoneA[j + 1];
* //     anotherphoneA[j] =anotherphoneA[j + 1];
* //     nationalityA[j] = nationalityA[j + 1];
* //     emailA[j] = emailA[j + 1];
* // }
* file.close();
* }
* }
* void subMenuBeforeMainMenu(string submenu)
* {
* string message = submenu + " Menu";
* cout << message << endl;
* cout << "---------------------" << endl;
* }
* void READ\_DATA\_from\_file\_of\_add\_user()
* {
* string line;
* fstream file;
* file.open("filenumber01.txt", ios::in);
* while (!(file.eof)())
* {
* getline(file, line);
* nameA[idx] = getfield(line, 1);
* genderA[idx] = getfield(line, 2);
* phoneA[idx] = getfield(line, 3);
* anotherphoneA[idx] = getfield(line, 4);
* nationalityA[idx] = getfield(line, 5);
* emailA[idx] = getfield(line, 6);
* showbase\_file\_record(nameA[idx], genderA[idx], phoneA[idx], anotherphoneA[idx], nationalityA[idx], emailA[idx]);
* idx = idx + 1;
* }
* }
* string getfield(string line, int field)
* {
* int space\_count = 1;
* string result\_fully\_line;
* for (int x = 0; x < line.length(); x++)
* {
* if (line[x] == ',')
* {
* space\_count = space\_count + 1;
* }
* else if (space\_count == field)
* {
* result\_fully\_line = result\_fully\_line + line[x];
* }
* }
* return result\_fully\_line;
* }
* void hotel()
* {
* cout << "1   luxury hotel " << endl
* << "  high class "
* << "per Room  Per day 10000  " << endl;
* cout << "2..... middle class  hotel " << endl
* << "  middle class "
* << "per Room  Per day 8000  " << endl;
* cout << "3..... normal class  hotel " << endl
* << "  normal class "
* << "per Room  Per day 7000  " << endl;
* cout << "choose one of these option " << endl;
* getch();
* }
* void select\_hoatl()
* {
* if (count\_student == 0)
* {
* cout << "PLease first data enter " << endl;
* getch();
* }
* else if (count\_student != y)
* {
* cout << "please first select package (no.4)" << endl;
* cout << "after do this, you are able to select hotal " << endl;
* getch();
* }
* else
* {
* string name;
* cout << "Enter the name " << endl;
* cin >> name;
* int c = 0;
* for (int x = 0; x < count\_student; x++)
* {
* if (name == nameA[x])
* {
* c = 2;
* cout << "Select one option from these hoatals" << endl;
* cout << "1 luxury hotal" << endl;
* cout << "2 Middle class hotal" << endl;
* cout << "3 Normal class hotal" << endl;
* cin >> option\_hotals;
* cout << " Enter the days" << endl;
* cin >> hotal\_days;
* cout << " Enter the members" << endl;
* cin >> hotal\_mens;
* cout << "Youn want A.C OR non A.C ROOM ?select from one these" << endl;
* cout << "1  YES" << endl;
* cout << "2   NO " << endl;
* cin >> count\_for\_AC\_or\_non\_AC;
* // if( count\_for\_AC\_or\_non\_AC != 1 ||  count\_for\_AC\_or\_non\_AC != 2 || option\_hotals != 1 || option\_hotals != 2 || option\_hotals != 3 )
* // {
* //  cout<<"you press wrong numbers"<<endl;
* //  getch();
* //  break;
* // }
* calulation\_of\_hotal\_deatil(hotal\_days, hotal\_mens, count\_for\_AC\_or\_non\_AC, option\_hotals);
* store\_data\_of\_hotal\_deatil\_into\_file();
* }
* }
* if (c == 2)
* {
* cout << "hotal  is successfully selected" << endl;
* }
* else if (c != 2)
* {
* cout << "you press wrong" << endl;
* }
* getch();
* }
* }
* void calulation\_of\_hotal\_deatil(int days, int mens, int number\_for\_AC\_or\_non\_AC, int choice)
* {
* hotal\_daysA[hotal\_index] = hotal\_days;
* hotal\_mensA[hotal\_index] = hotal\_mens;
* if (choice == 1 && number\_for\_AC\_or\_non\_AC == 1)
* {
* total\_price\_of\_hotal[hotal\_index] = price\_of\_luxury\_hotal \* hotal\_daysA[hotal\_index] \* hotal\_mensA[hotal\_index] + 5000;
* }
* else if (choice == 1 && number\_for\_AC\_or\_non\_AC == 2)
* {
* total\_price\_of\_hotal[hotal\_index] = price\_of\_luxury\_hotal \* hotal\_daysA[hotal\_index] \* hotal\_mensA[hotal\_index];
* }
* else if (choice == 2 && number\_for\_AC\_or\_non\_AC == 1)
* {
* total\_price\_of\_hotal[hotal\_index] = price\_of\_middle\_class\_hoatl \* hotal\_daysA[hotal\_index] \* hotal\_mensA[hotal\_index] + 3500;
* }
* else if (choice == 2 && number\_for\_AC\_or\_non\_AC == 2)
* {
* total\_price\_of\_hotal[hotal\_index] = price\_of\_middle\_class\_hoatl \* hotal\_daysA[hotal\_index] \* hotal\_mensA[hotal\_index];
* }
* else if (choice == 3 && number\_for\_AC\_or\_non\_AC == 1)
* {
* total\_price\_of\_hotal[hotal\_index] = price\_of\_normal\_class\_hoatl \* hotal\_daysA[hotal\_index] \* hotal\_mensA[hotal\_index] + 2500;
* }
* else if (choice == 3 && number\_for\_AC\_or\_non\_AC == 2)
* {
* total\_price\_of\_hotal[hotal\_index] = price\_of\_normal\_class\_hoatl \* hotal\_daysA[hotal\_index] \* hotal\_mensA[hotal\_index];
* }
* }
* void store\_data\_of\_hotal\_deatil\_into\_file()
* {
* fstream file2;
* file2.open("filenumber03.txt", ios::app);
* file2 << " Total price of hotal >>>" << total\_price\_of\_hotal[hotal\_index];
* file2 << " days>>>" << hotal\_daysA[hotal\_index];
* file2 << " mens>>>" << hotal\_mensA[hotal\_index];
* file2 << endl;
* file2.close();
* hotal\_index++;
* }
* void changr\_the\_price\_of\_hotal()
* {
* cout << "Set the price of luxury hotal       " << endl;
* cin >> m;
* cout << "Set the price of middle class hotal       " << endl;
* cin >> n;
* cout << "Set the price of normal hotal       " << endl;
* cin >> o;
* price\_of\_luxury\_hotal = m;
* price\_of\_middle\_class\_hoatl = n;
* price\_of\_normal\_class\_hoatl = o;
* cout << "Prices of hotal are successfully changed" << endl;
* getch();
* }
* void shange\_the\_price\_of\_packages()
* {
* cout << "Set the price of Golden package       " << endl;
* cin >> p;
* cout << "Set the price of normal package       " << endl;
* cin >> q;
* cout << "Set the price of  lower package       " << endl;
* cin >> r;
* package\_1\_price = p;
* package\_2\_price = q;
* package\_3\_price = r;
* cout << "Prices of packages are successfully changed" << endl;
* getch();
* }
* void view\_hotal()
* {
* system("cls");
* subMenuBeforeMainMenu("Manu of  hotal");
* cout << "1 luxury hotal" << endl;
* cout << price\_of\_luxury\_hotal << "Rs." << endl;
* cout << "2 Middle class hotal" << endl;
* cout << price\_of\_middle\_class\_hoatl << "Rs." << endl;
* cout << "3 Normal class hotal" << endl;
* cout << price\_of\_normal\_class\_hoatl << "Rs." << endl;
* getch();
* }
* string showbase\_file\_record(string name1, string gender1, string phone1, string an\_phone, string nationality1, string email1)
* {
* cout << "CUSTOMER " << count\_for\_show\_user\_deatil\_from\_file + 1 << endl;
* ;
* cout << name1 << endl;
* cout << gender1 << endl;
* cout << phone1 << endl;
* cout << an\_phone << endl;
* cout << nationality1 << endl;
* cout << email1 << endl;
* cout << endl;
* count\_for\_show\_user\_deatil\_from\_file++;
* }
* string getfield1(string line1, int field1)
* {
* int space\_count1 = 1;
* string result\_fully\_line1;
* for (int x = 0; x < line1.length(); x++)
* {
* if (line1[x] == ',')
* {
* space\_count1 = space\_count1 + 1;
* }
* else if (space\_count1 == field1)
* {
* result\_fully\_line1 = result\_fully\_line1 + line1[x];
* }
* }
* return result\_fully\_line1;
* }
* void READ\_DATA\_from\_file\_of\_packages()
* {
* string line\_used;
* string a, b, c;
* fstream file1;
* file1.open("filenumber02.txt", ios::in);
* while (!(file1.eof)())
* {
* getline(file1, line\_used);
* a = getfield1(line\_used, 1);
* b = getfield1(line\_used, 2);
* c = getfield1(line\_used, 3);
* show\_record\_packages(a, b, c);
* idx = idx + 1;
* }
* getch();
* }
* void show\_record\_packages(string tt, string ds, string ms)
* {
* cout << "PACKAGE DEATIL OF CUSTOMER" << count\_for\_show\_package\_deatil\_from\_file + 1 << endl;
* cout << tt << endl;
* cout << ds << endl;
* cout << ms << endl
* << endl;
* count\_for\_show\_package\_deatil\_from\_file++;
* }
* string getfield2(string line2, int field2)
* {
* int space\_count2 = 1;
* string result\_fully\_line2;
* for (int x = 0; x < line2.length(); x++)
* {
* if (line2[x] == ',')
* {
* space\_count2 = space\_count2 + 1;
* }
* else if (space\_count2 == field2)
* {
* result\_fully\_line2 = result\_fully\_line2 + line2[x];
* }
* }
* return result\_fully\_line2;
* }
* void READ\_DATA\_from\_file\_of\_hotals()
* {
* string line\_used\_for\_hotal;
* string a, b, c;
* fstream file2;
* file2.open("filenumber02.txt", ios::in);
* while (!(file2.eof)())
* {
* getline(file2, line\_used\_for\_hotal);
* a = getfield1(line\_used\_for\_hotal, 1);
* b = getfield1(line\_used\_for\_hotal, 2);
* c = getfield1(line\_used\_for\_hotal, 3);
* show\_record\_hotals(a, b, c);
* idx = idx + 1;
* }
* getch();
* }
* void show\_record\_hotals(string tth, string dsh, string msh)
* {
* cout << "HOTAL  DEATIL OF CUSTOMER" << count\_for\_show\_hotal\_detail + 1 << endl;
* cout << tth << endl;
* cout << dsh << endl;
* cout << msh << endl
* << endl;
* count\_for\_show\_hotal\_detail++;
* }
* bool signIn(string username, string password)
* {
* for (int i = 0; i < usercount; i++)
* {
* if (username == usernames[i] && password == passwords[i])
* {
* return true;
* }
* }
* return false;
* }
* void addUsers(string username, string password)
* {
* usernames[usercount] = username;
* passwords[usercount] = password;
* usercount++;
* }
* void viewUsers()
* {
* cout << left << setw(20) << "Usernames" << setw(20) << "Passwords" << endl;
* for (int i = 0; i < usercount; i++)
* {
* cout << left << setw(20) << usernames[i] << setw(20) << passwords[i] << endl;
* }
* }
* void writeFile(string username, string password)
* {
* fstream file;
* file.open("UserDetails.txt", ios::app);
* file << username << "\n";
* file << password << "\n";
* file.close();
* }
* bool isValidUsername(string username)
* {
* for (int i = 0; i < usercount; i++)
* {
* if (username == usernames[i])
* {
* return false;
* }
* }
* return true;
* }
* int takeChoice()
* {
* int choice;
* cout << "01. SignUp" << endl;
* cout << "02. SignIn" << endl;
* cout << "00. Exit" << endl;
* cin >> choice;
* return choice;
* }
* void loadDataFromFile()
* {
* string username, password;
* fstream file;
* int counter = 0;
* file.open("UserDetails.txt", ios::in);
* while (getline(file, username) && getline(file, password))
* {
* usernames[usercount] = username;
* passwords[usercount] = password;
* usercount++;
* }
* file.close();
* }
* void bill()
* {
* string name;
* cout<<"enter the name which you want to see bill"<<endl;
* cin>>name;
* int n = 0;
* for(int x =0 ; x<count\_student;x++)
* {
* if(name == nameA[x])
* {
* n =2;
* cout<<nameA[x]<<endl;
* cout<<nationalityA[x]<<endl;
* cout<<genderA[x]<<endl;
* cout<<phoneA[x]<<endl;
* cout<<anotherphoneA[x]<<endl;
* cout<<emailA[x]<<endl;
* cout<<"<<<<<package<<<<"<<endl;
* cout<<daysA[x]<<"days"<<endl;
* cout<<mensA[x]<<"mens"<<endl;
* cout<<"total price"<<total\_price\_of\_packages[x]<<endl;
* cout<<"<<<<<hotal<<<<"<<endl;
* //  cout<<hotal\_daysA<<"days"<<endl;
* //  cout<<hotal\_mensA<<"mens"<<endl;
* cout<<"total price"<<total\_price\_of\_hotal[x]<<endl;
* }
* }
* if(n == 2)
* {
* cout<<"bill is generated"<<endl;
* getch();
* }
* else
* {
* cout<<"you press wrong"<<endl;
* getch();
* }
* }

**Other General Instructions**

1. Attach 2 Rubrics after the documentation. Your evaluation will be done according to the rubrics.
2. Rubric format is provided below.
3. Make sure all components in the rubric are present in your code and documentation.
4. Rubric should be on single page (Do not spread it to two pages)
5. Final Documentation should be submitted in Tape Binding.

**Student Reg. No. :**   **Student Name.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **A-Extensive Evidence** | **B-Convincing Evidence** | **C-Limited Evidence** | **D-No Evidence** |
| Documentation Formatting  **Grade:** | All the documentation meets all the criteria. | Documentation is well formatted but some of the criteria is not fulfilled. | Documentation is required a lot of improvement. | Documentation is not Available |
| **Documentation Formatting Criteria:** In **Binder**, **Title** Page, **Header**-Footers, Font **Style**, Font **Size** all are all consistence and according to given **guidelines**. Project **Poster** is professionally design and well presented | | | | |
| Documentation Contents  **Grade:** | Documentation includes all of the criteria. | Documentation meet more than 80% of the criteria given. | Documentation meet more than 50% of the criteria. | When the documentation meet less than 50% of the criteria. |
| **Documentation Contents Criteria:** **Title** Page - **Table** of Contents - Project **Abstract** - **Functional** Requirements - **Wire** Frames –**Data Flow** Diagram-**Data** Structure (Arrays)-**Function** Headers and Description -Project **Code.** - **Weakness** in the Project and **Future** Directions. - **Conclusion** and What your **Learn** from the Project and Course and What is your **Future** Planning. | | | | |
| Project Complexity  **Grade:** | Project has at least 2 user’s types and each user has at least 5 functionalities. | Project complexity meet 80% criteria given in extensive evidence | Project complexity meet 50% criteria given in extensive evidence | Project complexity meet less than 50% criteria given in extensive evidence |
| Code Style  **Grade:** | All Code style criteria is followed | All code style criteria followed but some improvements required | lot of improvements required in coding style. | **Did not follow** code style, |
| **Code Style Criteria:**  Consistent code style. Code is well indented. Variable and Function names are well defined.  White Spaces are well used. Comments are added. | | | | |
| Code Documentation Mapping  **Grade:** | Code and documentation is synchronized. | Code and documentation does not synchronized at **some** places | Code and documentation does not synchronized at **many** places | Code and documentation **does not** synchronized. |
| Data Structure (Arrays)  **Grade:** | Data structure is sufficient for the project requirements | Data Structure is sufficient but require improvement to meet project requirements. | Data structure is not sufficient and need a lot of improvement | Data Structure is not properly identified and declared. |
| Modularity  **Grade:** | Meet all Modularity criteria | Meet all Modularity criteria but at some places it is missing | Do not sufficiently meet the modularity criteria. | No modularity or very minimum modularity. |
| **Modularity criteria:** Functions are defined for each major feature. Functions are independent (identify from parameter list and return types). | | | | |
| Validations  **Grade:** | Validations on all number type inputs are applied | Validations are applied but at some places it is missing. | Validations are missing at lot of places | No Validations are used |
| File Handling  **Grade:** | Separate files for separate data. Data in csv format | File handing require some improvements | File handing require a lot of improvements | Not implemented |
| Aesthetics of the User Interface  **Grade:** | UI is presentable. Proper coloring, Headers and clear screen is done | UI require some improvements | UI require a lot of improvements | Not implemented |
| Presentation and Demo  **Grade:** | Presentation and Demo was 100% working | Presentation and Demo require some improvements | Presentation and Demo require a lot of improvements | Presentation was not ok and Demo was not working |
| Student Understanding with the Code.  **Grade:** | Student has complete understanding how the code is working and knows the concept. | Student has good understand but some place he does not know the concepts | Student has a very little understand and lack the major concepts. | Student does not have any level of understanding of the code. |

|  |  |
| --- | --- |
| **Checked by:** |  |
| **Comments:** |  |